

Instruktion för instuderingsmaterial

Fakta

Kurs:	Systemutvecklare C/C++ Extended 2024
Klass:	SUVx24
Teknikområde:	Programspråket C, Arrayer, break & continue

Learning Target

Programmering i C och C++

Hur använder man `arrayer` som lagringsplats för liknande värden som hör ihop?

När vi bygger program händer det ibland att data inte består av bara ett värde, utan är en serie av datavärden som hör ihop. Då får vi skapa lika många variabler som datavärden, men det är ibland okänt hur många datavärden som serien innehåller. För att slippa skapa onödigt många variabler (för säkerhets skull) blir lösningen på detta att använda **arrayer**. Då kan vi samla serien med data till ett och samma (variabel)namn och sedan använda **index** för att hitta rätt data i samlingen.

Introduktion

Det finns sätt att styra `for`- och `while`-loopar att inte utföra precis alla loopens möjliga steg, utan antingen bryta av loopandet eller att hoppa över ett steg. Detta är möjligt med `break` och `continue`. Break har vi redan stött på i `switch..case`-satsen.

`Arrayer` (listor) byggs upp som en serie med lådor där varje låda har en position med ett index. Alla lådorna i en viss **array** måste innehålla värden av samma (data)typ. Ofta hanterar vi `arrayer` med hjälp av `for`-loopar, men ibland även med `while`.

Material

Här nedan hittar du ditt självstudiematerial. Du ska studera materialet i den ordning det listas för att du på bästa sätt ska tillgodogöra dig materialet.

Instruktion för instuderingsmaterial

Learning Target

Hur styr man `for`- och `while`-loopar med `break` och `continue`?

Hur

Läs nedanstående textavsnitt och de specificerade underdelarna. Studera kodexemplen för att förstå principerna.

Först tittar vi på hur `break` och `continue` kan hjälpa oss att styra `for`- och `while`-loopar på liknande sätt som i `switch..case`.

Textavsnitt från [w3schools](https://www.w3schools.com/c/c_break_continue.asp)

C Break and Continue

- Hur avbryter du en `for` loop i förtid?
- Hur kan du hoppa över en eller flera steg i loopen?
- Fungerar det olika på `for` respektive `while`-loop?

Textavsnitt från [geeksforgeeks.org](https://www.geeksforgeeks.org/c-break-and-continue-statements/)

GeeksForGeeks har också information om `break` och `continue`.

Difference between break and continue statement in C

Läs avsnittet om `break` och `continue`.

- Prova kodexemplen
- Studera den avslutande tabellen.
 - När kan man använda `break` förutom med `for` och `while`?

Textavsnitt från [programiz.com](https://www.programiz.com/c-programming/c-break-and-continue)

C break and continue

Här finns det förklarande bilder till samma tema på `break` och `continue`.

- Läs avsnitten *C break* och *C continue*
- Titta gärna på videon längst ned på sidan.

Instruktion för instuderingsmaterial

Learning Target

Hur använder man `arrayer` som lagringsplats för liknande värden som hör ihop?

Hur

Läs nedanstående textavsnitt och de specificerade underdelarna. Studera kodexemplen för att förstå principerna.

Textavsnitt från [w3schools](#)

C Arrays

- Läs avsnitten Arrays, Array Size, Real-Life Example, Multidimensional Arrays.
- Vilka speciella symboler behöver vi ta med i samband med arrayer?
 - Vad symboliserar de?
- Hur ser en initiering av en array ut?
- Hur byter man värde på en viss position i arrayen?
- Hur går man genom en arrays innehåll?
- Kan man ta reda på storleken av en array, dvs hur många positioner som det går att lagra data i?
 - Är det något särskilt vi måste tänka på när vi använder `sizeof`-funktionen?
- Går det att ändra storleken på en array när den är skapad?
 - Vad måste man göra om man vill ändra på en arrays storlek?
- Vad händer om vi försöker bestämma storleken på en tom array?
- Hur initierar man en tvådimensionell array?
 - Hur går man genom arrayens värden?
 - Går det att deklarera/initiera fler dimensioner än två?

C Strings

Läs avsnitten *Strings*, *Special Characters*, *String Functions*

- Vad är Strings och varför vill vi gärna använda dem?
- Varför måste vi använda arrayer för att hantera Strings i C?
- Hur kan vi plocka ut och använda delar av en String?
- Vilka symboler använder vi för att beteckna en String?
- Vad måste vi tänka på att hantera i slutet av en String?
- Hur skriver vi ut vissa specialtecken utan att de tolkas som styrtecken i koden?
- Vad är *Escape Character*?

Instruktion för instuderingsmaterial

Textavsnitt från [geeksforgeeks.org](https://www.geeksforgeeks.org)

GeeksForGeeks har också information om `break` och `continue`.

C Arrays

Här finns det bilder som stöd till förklaringen hur arrayer fungerar.

- Läs alla delarna fram till och med B. Three-Dimensional Array in C.
- Hoppa över texten från Relationship between Arrays and Pointers och framåt så länge.

Strings in C

Mer om strängar (Strings).

- Hur läser man in en sträng av ord med mellanslag?
- Hur kan man läsa av längden på en sträng?
- Vilka smarta funktioner finns i standardbiblioteket `<string.h>`?

Textavsnitt från [programiz.com](https://www.programiz.com)

C Arrays

Programiz förklaring till arrayer.

- Läs avsnitten *C arrays* och *C Multidimensional Arrays*
- Vad händer när man försöker läsa från positioner utanför en array?
- Titta gärna på videon längst ned på sidan.

C Programming Strings

Ännu mer förklaringar till hur strängar (Strings) fungerar.

- Vad gör `fgets()` och `puts()`?
- Hoppa över från *Passing Strings to Functions*.
- Titta gärna på videon längst ned på sidan.